



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2017

Probing many-body localization with neural networks

Schindler, Frank ; Regnault, Nicolas ; Neupert, Titus

DOI: <https://doi.org/10.1103/PhysRevB.95.245134>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-141673>

Journal Article

Published Version



The following work is licensed under a Creative Commons: Attribution 4.0 International (CC BY 4.0) License.

Originally published at:

Schindler, Frank; Regnault, Nicolas; Neupert, Titus (2017). Probing many-body localization with neural networks. Physical review. B, 95:245134.

DOI: <https://doi.org/10.1103/PhysRevB.95.245134>

Probing many-body localization with neural networks

Frank Schindler,¹ Nicolas Regnault,² and Titus Neupert¹

¹*Department of Physics, University of Zurich, Winterthurerstrasse 190, 8057 Zurich, Switzerland*

²*Laboratoire Pierre Aigrain, Département de physique de l'ENS, Ecole normale supérieure, PSL Research University, Université Paris Diderot, Sorbonne Paris Cité, Sorbonne Universités, UPMC Univ. Paris 06, CNRS, 75005 Paris, France*

(Received 11 April 2017; published 26 June 2017)

We show that a simple artificial neural network trained on entanglement spectra of individual states of a many-body quantum system can be used to determine the transition between a many-body localized and a thermalizing regime. Specifically, we study the Heisenberg spin-1/2 chain in a random external field. We employ a multilayer perceptron with a single hidden layer, which is trained on labeled entanglement spectra pertaining to the fully localized and fully thermal regimes. We then apply this network to classify spectra belonging to states in the transition region. For training, we use a cost function that contains, in addition to the usual error and regularization parts, a term that favors a confident classification of the transition region states. The resulting phase diagram is in good agreement with the one obtained by more conventional methods and can be computed for small systems. In particular, the neural network outperforms conventional methods in classifying individual eigenstates pertaining to a single disorder realization. It allows us to map out the structure of these eigenstates across the transition with spatial resolution. Furthermore, we analyze the network operation using the dreaming technique to show that the neural network correctly learns by itself the power-law structure of the entanglement spectra in the many-body localized regime.

DOI: [10.1103/PhysRevB.95.245134](https://doi.org/10.1103/PhysRevB.95.245134)

I. INTRODUCTION

Artificial neural networks are routinely employed for data classification. They are useful when features distinguishing one class of data from another are unknown or unwieldy. A neural network can learn such features from examples, i.e., a set of labeled training data. In physics, the application of neural networks, and machine learning in general, to many-body quantum mechanics is a novel and burgeoning field of research [1]. Currently, there are three main lines of pursuit: the application of machine learning to the problem of classifying various phases of matter [2–9], accelerating material searches and design [10–13], and the quest to encode quantum mechanical states in structures mimicking the setup of a neural network [14–16]. This work is concerned with the first kind of approach. Most previous studies have considered the identification of phases and phase transitions by training neural networks on a large set of prototype configurations. Here, we instead use entanglement spectra [17], which in recent years emerged as a powerful tool to characterize a plethora of physical systems, and have been employed for a neural network based detection of phase transitions in Ref. [8].

We apply neural network based phase classification to a fundamental question in quantum statistical physics, namely, the distinction between systems that obey the eigenstate thermalization hypothesis (ETH) and those violating it. According to the ETH, local observables in a typical many-body eigenstate should take the values that pertain to the observables in a thermal ensemble, with the whole system acting as a heat bath for its subsystems in the thermodynamic limit. A well-studied class of systems that violate the ETH are those exhibiting many-body localization (MBL) [18–25], meaning that partial memory of initial conditions is preserved for infinite times. Due to this property, which is intimately related to the emergence of an extensive number of integrals of motion [23,26–28], MBL systems have been envisioned as particularly

robust quantum memories [29]. Here, we study the Heisenberg chain in a random field as a simple model for MBL. At strong disorder, the model is in the MBL regime, whereas it satisfies the ETH if disorder is weak. Several measures or quantities allow a well-controlled quantitative distinction of thermal and localized regimes. They have been used to study the ETH-MBL transition in finite size numerical simulations, in particular for an extensive analysis of the Heisenberg model in a random field. These characterizing quantities include energy level statistics [30–35], level statistics [25,36] as well as density of states [37] analyses of the entanglement spectrum and studies of the distribution of the entanglement entropy over a region of energy eigenstates [18,38–43]. Necessarily, these methods rely on a physical understanding of the nature of either regime or of the transition. The neural network based method for identifying the ETH-MBL transition that we present here requires only that the information for distinguishing the ETH from the MBL regime is—in some form—contained in the entanglement spectrum. This is useful in particular in situations where the physical characteristics of a phase are not fully understood, as one may certainly argue to be the case for MBL [44,45]. Thus, the neural network approach also allows for the possibility of finding ways of characterizing the phase transition beyond established methods, with the network's architecture providing a variational ansatz for a classification criterion.

We use the network to classify the entanglement spectra of *all eigenstates* of the Heisenberg chain, which are obtained by exact diagonalization, in particular at finite energy density (note that Ref. [8] has characterized the transition using *ground state* properties of the disordered Heisenberg chain via a neural network-based approach of classifying entanglement spectra). For a specific disorder configuration, this allows, for instance, to trace the evolution of individual ETH states deep in the MBL regime [40,46–48]. We achieve this by considering the spectra from multiple real-space entanglement cuts as input for the

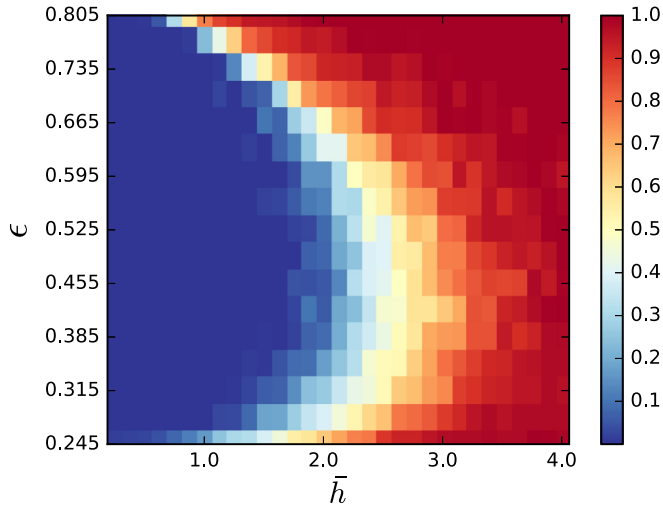


FIG. 1. Phase diagram of the Heisenberg chain with Hamiltonian (5) obtained from the neural network ansatz in Eq. (7) trained with cost function (8) on entanglement spectra obtained from an exact diagonalization on $N = 16$ sites. The plot shows the average confidence for the MBL phase over 40 realizations of disorder as a function of the absolute maximal values of the random magnetic field \bar{h} , spaced with $\Delta\bar{h} = 0.125$, and for eigenstates belonging to different rescaled energies $\epsilon = (E - E_{\min})/(E_{\max} - E_{\min})$. Compared to Ref. [18] where a similar plot was obtained with better-controlled, yet more sophisticated methods, we have used smaller systems and fewer disorder realizations.

neural network. By averaging over disorder realizations, we obtain a phase diagram (Fig. 1) that indicates the location of the ETH-MBL transition as a function of energy density and disorder strength. It is in good agreement with results obtained using conventional methods [18,36,43,49].

This paper is organized as follows. In Sec. II, which may also be read as a short introduction to neural networks, we introduce the general setup of the network used here, suited for binary classification of data. Subsequently, in Sec. III, we review the Heisenberg spin chain in a random field, and define the entanglement spectrum. We then discuss the type of input data as well as the network architecture used for classifying entanglement spectra as MBL or ETH in Sec. IV. In Sec. V, we present our results and compare them to existing methods.

II. NEURAL NETWORKS FOR BINARY CLASSIFICATION

An artificial neural network is an alternating sequence of affine linear maps and nonlinear functions that are successively applied to input data x giving output y . Each pair of maps in this sequence is a *layer* of the network. Let the target space of the α th layer of the network have dimension $n_{\alpha+1}$, corresponding to $n_{\alpha+1}$ neurons. In this work, we focus on binary classification, where we want to learn a map $f(x)$ from the data set $\{x\}$, represented by vectors x of dimension n_1 , to the discrete target set $\{(0,1), (1,0)\}$. This representation of the target set is somewhat arbitrary—here, we choose *one-hot* vectors, i.e., vectors with a single nonzero element. Their entries are interpreted as the neurons of the output layer.

The network setup described above now implements a trial map \hat{f} , which should approximate the unknown map f as good

as possible. One important difference is that while the target space of f is discrete, that of \hat{f} is continuous. This allows for smooth convergence of \hat{f} to f . To achieve this, we first train the network by adjusting its parameters to gradually improve its performance on a *training set* which is labeled, i.e., for which the output of f is known to be either (0,1) or (1,0) for each x . We then apply the network to a *testing set* to evaluate how well it generalizes to classify data that it has not seen before. It is essential to avoid overfitting: with a large number n_2 of neurons, the network will learn not only the general rules by which the data can be identified as pertaining to the MBL or ETH regimes. Rather, it will also pick up nonuniversal features, such as noise specific to the training data set that was used. To improve the generalization capability of the network at this stage, we employ *cross-validation*: we first obtain the training and testing sets by randomly subdividing a large set of labeled data into two parts of equal size, and then average the trained network's output (when applying it to previously unlabelled data) over multiple such training runs.

We now describe the full action of the network on the input data. In the first layer, the input vectors x of dimension n_1 are mapped to a space of dimension n_2 via an affine linear map $x \mapsto V^{(2,1)}x + a^{(2)}$, followed by the application of a nonlinear *activation function* g_2 (the nonlinearity of which is required in order to be able to approximate arbitrary maps f), so that the full action of the first layer may be written as $x \equiv x^{(1)} \mapsto x^{(2)} = g_2(V^{(2,1)}x + a^{(2)})$. Here, $V^{(2,1)}$ is a $n_2 \times n_1$ matrix, and matrix-vector multiplication between $V^{(2,1)}$ and x is implied here as well as below. Each entry of the resulting vector $x^{(2)}$ can be interpreted as the output of an individual neuron, of which there are n_2 in total. In general, the first layer is followed by further layers, each of which implements the map

$$x^{(\alpha)} \mapsto x^{(\alpha+1)} = g_{\alpha+1}(V^{(\alpha+1,\alpha)}x^{(\alpha)} + a^{(\alpha+1)}). \quad (1)$$

The elements of the rows in the matrix $V^{(\alpha+1,\alpha)}$ are called the *weights* of the respective neuron, and the corresponding element of the vectors $a^{(\alpha+1)}$ are referred to as its *bias*. All layers but the last one are called *hidden layers*. If there are h hidden layers, $x^{(h+2)} = y$ is the two-component output vector. In the networks we use, all vectors, matrices, and numbers are real.

In the following, we will use a network with $h = 1$, built from the activation functions $g_2 = \text{ReLU}$ and $g_3 = \text{Softmax}$, defined as

$$\begin{aligned} \text{ReLU}_i(x) &= x_i \theta(x_i), \\ \text{Softmax}_i(x) &= \frac{e^{-x_i}}{\sum_j e^{-x_j}}, \end{aligned} \quad (2)$$

which are applied componentwise on their vector-valued argument, and the indices i, j run over these components. The projections of the Softmax output onto the target set vectors sum up to 1 and can be interpreted as the confidences with which the network allocates the input data to the respective class. For a schematic representation of our network, see Fig. 2.

In order for \hat{f} to approximate f , we need to tune the parameters of the network, i.e., the weights and biases, to minimize the discrepancy encoded in an appropriately chosen error functional $\text{Cost}(\hat{f}, f)$. The common choice suited for a

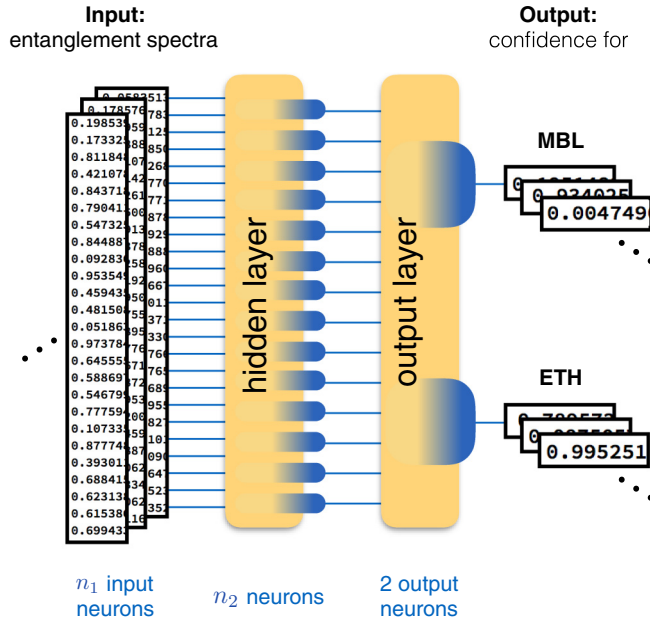


FIG. 2. Schematic setup of the neural network used to map entanglement spectra to the confidence with which they are classified as either belonging to the ETH or MBL regime. This map, which is explicitly given by Eq. (7), can be interpreted as the action of a hidden layer of neurons on the input data, followed by an output layer of two neurons which correspond to the two options of classification. Note that our choice of a Softmax activation function for the output layer implies that the confidences for ETH and MBL sum up to 1.

Softmax output layer is the cross entropy

$$\text{Cost}(\hat{f}, f) = - \sum_{x \in \{x\}} \sum_{i=1}^2 f_i(x) \ln \hat{f}_i(x). \quad (3)$$

In training, we then hope to find the global minimum of this functional. One starts from, e.g., randomly initialized weights and biases, which we jointly denote as X_0 , and then successively applies gradient descent to the weights and biases at step n to obtain those at $n + 1$ as

$$X_{n+1} = X_n - \lambda \frac{\partial}{\partial X_n} \text{Cost}, \quad X_n \in \{V_{ij}^{(\alpha+1, \alpha)}, a_i^{(\alpha)}\}. \quad (4)$$

The step size λ should neither be too large (otherwise minima are overlooked), nor too small (otherwise convergence is slow and it becomes harder to escape from local minima). A parameter such as λ , which is not changed during training, but rather determines how we train, is called a hyperparameter. Here, we fix λ empirically by requiring optimal minimization of the error on the *training* data. Each such iteration $X_n \mapsto X_{n+1}$ of gradient descent is called a *training step*. Since it is too cumbersome to evaluate the error functional for large training sets, we employ stochastic gradient descent; for each iteration, one randomly chooses a relatively small subset of $\{x\}$ as training data. Note that from the point of view of variational calculus, a neural network just corresponds to a shrewd and economic choice of ansatz for minimizing the functional (3).

III. MANY-BODY LOCALIZATION IN THE HEISENBERG CHAIN AND ENTANGLEMENT SPECTRUM

As a toy model for MBL, we study the Heisenberg Hamiltonian in a random field in z direction,

$$H = J \sum_{r=1}^{N-1} S_r \cdot S_{r+1} + \sum_{r=1}^N h_r S_r^z, \quad (5)$$

on an N -site chain of spin-1/2 degrees of freedom with open boundary conditions. Here, $S = \frac{1}{2}\sigma$ acts on the spin on a given site, with σ the vector of Pauli matrices, and the h_r , $r = 1, \dots, N$, are static random external fields taken from a uniform distribution in the interval $[-\bar{h}, \bar{h}]$. In the following, we will set $J = 1$. The system is integrable for $\bar{h} = 0$. System realizations with $\bar{h} \ll 1$ are in a thermalizing (ETH) regime. System realizations with $\bar{h} \gg 1$ are in an MBL regime. Both regimes are characterized by different energy level statistics: the ETH regime exhibits level repulsion obeying the Gaussian orthogonal ensemble (GOE) for the Heisenberg Hamiltonian of Eq. (5). On the other hand, the energy spectrum in the MBL regime has Poisson level statistics.

In between the two limits, the behavior of a specific system being either ETH or MBL depends on the specific disorder realization and the eigenstate that is considered. Averaging over disorder realizations removes these dependencies, but the transition between ETH and MBL regimes may still depend on the energy density at which the system is probed, which amounts to the existence of a many-body mobility edge. We will assume that at $\bar{h} = 0.25$ and 12.0 almost all eigenstates belong to the ETH or MBL regime, respectively.

A characteristic that has been shown to discriminate between ETH and MBL regimes is the entanglement spectrum. It is defined as follows. Consider the reduced density matrix ρ_A of a system in the pure state $|\Psi\rangle$ obtained by subdividing the Hilbert space into two parts, A and B , and tracing out the degrees of freedom of B ,

$$\rho_A = \text{Tr}_B |\Psi\rangle \langle \Psi| \equiv e^{-H_e}. \quad (6)$$

The last equality defines the entanglement Hamiltonian H_e . Here we are interested in a real-space cut separating regions A and B such that all lattice sites $r \leq N_A$, for some $0 < N_A < N$, are in A , and B is the complement of A . The spectrum of H_e is called the entanglement spectrum, and contains information about the nature of $|\Psi\rangle$.

Several possibilities have been explored to determine from the entanglement properties whether a state $|\Psi\rangle$ at finite energy density and fixed disorder shares the character of the MBL or ETH regime. (i) The ‘‘Schmidt gap’’ $\lambda_1(\rho_A) - \lambda_2(\rho_A)$, where $\{\lambda_j(\rho_A); \lambda_j \geq \lambda_{j+1}\}$ denotes the spectrum of ρ_A . Being the difference of the two largest eigenvalues of the density matrix, i.e., of the square of the two largest coefficients in the Schmidt decomposition of the system into A and B , it is nearly 0 for mixed ρ_A , typical for the ETH regime, and approximates 1 for almost pure ρ_A , characteristic of the MBL phase [50]. (ii) ETH states have volume-law entanglement scaling, while MBL states have area-law entanglement scaling. To discriminate between the two in a one-dimensional system, one computes the entanglement entropy $S(N_A)$ as a function of N_A . Extensive scaling of $S(N_A)$ with N_A is expected in the ETH regime,

while $S(N_A)$ is constant over different values of N_A in the MBL regime. (iii) The standard deviation σ_E of a sample of entanglement entropies calculated from eigenstates in a range of energies $[E, E + \Delta E]$. Within either phase, σ_E is small, while near the transition, where we find both MBL-like and ETH-like states in the energy interval that is probed, σ_E is enhanced [39–41, 43]. (iv) The level spacings in the entanglement spectrum follow distinct statistical distributions in the ETH and MBL regimes. A statistical analysis of the level distributions thus allows to identify the nature of individual eigenstates [25, 36].

The power of the neural-network based approach of classifying entanglement spectra as ETH or MBL that we pursue here is that it does not require any *a priori* knowledge of such criteria. Indeed, the neural network is expected to learn them by itself from the training by examples. In Sec. V, we compare its performance with (i) and (iii), as well as with the energy level statistics.

IV. TRAINING DATA AND NETWORK ARCHITECTURE

We train with a single-hidden-layer neural network aimed at binary classification of entanglement spectra for eigenstates obtained from the exact diagonalization of Hamiltonian (5). For a N -site chain, there are $|\{x\}| = 2^N$ eigenstates. Notice, however, that the total spin projection in z direction measured by the operator $S_{\text{tot}}^z = \sum_{r=1}^N S_r^z$ commutes with the Hamiltonian (5), corresponding to a global spin rotation symmetry. In the following, we focus on eigenstates in the $S_{\text{tot}}^z = 0$ sector. In the $S_{\text{tot}}^z = 0$ subspace, we are thus left with $|\{x\}| = \binom{N}{N/2}$ states, where we only use chains with even N here.

For a cut of size N_A on a N -site chain, there are $n_1 = 2^{N_A}$ levels in each entanglement spectrum. We can further make use of $S_{\text{tot},A}^z = \sum_{r=1}^{N_A} S_r^z$ to block-diagonalize the entanglement Hamiltonian. From now on, we focus on the largest block, e.g., with $S_{\text{tot},A}^z = 0$ if N_A is even. In this subspace, the entanglement spectrum has length $n_1 = \binom{N_A}{N_A/2}$, where $\lfloor N_A/2 \rfloor$ is the integer part of $N_A/2$. For training, we additionally leave out the eigenstates at very low and high energies, which are known to deviate substantially from the general trend of the given phase (concretely, we remove the 10% highest and 10% lowest energy states). After obtaining the reduced density matrix ρ_A we need to take the logarithm of its eigenvalues to arrive at the entanglement spectrum according to Eq. (6), a procedure, which is prone to numerical errors due to finite machine precision. Hence, we use only the first half (that is, the lower-lying half) of each entanglement spectrum for training, since in both the MBL and ETH regimes, the second half generically consists only of ρ_A eigenvalues which are smaller than 10^{-16} and therefore cannot contain any information. Note that the exact size of the part of the entanglement spectrum we train with is irrelevant, we checked that if we instead choose 1/3 or 2/3 of it the resulting phase diagram does not change.

We choose the activation functions as in Eq. (2), so that the full action of the network is

$$\hat{f}(x) = \text{Softmax}[V^{(3,2)} \text{ReLU}(V^{(2,1)} x + a^{(2)}) + a^{(3)}], \quad (7)$$

where $V^{(2,1)}$ and $V^{(3,2)}$ are $n_2 \times n_1$ and $2 \times n_2$ weight matrices, respectively, while $a^{(2)}$ and $a^{(3)}$ are the corresponding n_2 and

two-dimensional bias vectors. Here, $n_1 = \binom{N_A}{\lfloor N_A/2 \rfloor}$, and n_2 is a free parameter. In fact, we will take n_2 to have a relatively large value, of the order of 10^3 . In doing so, the network will become prone to overfit. To avoid overfitting, we employ two strategies (in addition to cross-validation, discussed in Sec. II).

(1) *Dropout regularization*. In each training step, we only train with half of the hidden layer neurons, which are randomly chosen each time, effecting the replacement $V^{(2,1)} \rightarrow P V^{(2,1)}$ in Eq. (7), where P projects onto a random subset of size $n_2/2$ of the hidden layer units. This prevents successive build-up of neuronal weight configurations adjusted to nonuniversal properties of the training data, and speeds up the convergence of the testing set error.

(2) *Weight decay*. During training, weights that have attained nonzero values at some point, but are no longer actively contributing to the minimization of the error function, should decay to zero in subsequent training steps. This can be achieved by adding a term $-\mu X_n$ on the right-hand side of Eq. (4), which corresponds to an augmentation of the error functional (3) by the term $\mu |X|^2$, where $|\cdot|$ denotes the l_2 norm of vectors and matrices. Here, we will apply weight decay only to the hidden layer weights $V^{(2,1)}$, since only this preserves a certain reparametrization symmetry of the network [51].

With these regularization methods, the number of training steps does not need to be fine-tuned as long as it is large enough. Independent of system size we find that a network with the described architecture classifies samples of testing data, which have no overlap with the training data, very successfully with an accuracy of $\eta = 1$, where η is the ratio of correctly identified spectra to all spectra in the testing set. Note that this is the case independent of whether we train and test with entanglement spectra obtained from the same or different disorder realizations.

However, being able to distinguish between the pure ETH and MBL regimes alone, at $\bar{h} = 0.25$ and $\bar{h} = 12.0$, respectively, is not enough to uniquely determine the classification strategy learned by the neural network. In order to make the predictions for the transition region reliable, we introduce *confidence optimization*. The network should classify entanglement spectra at intermediate \bar{h} values with maximal confidence. Note that this *does not require any prior knowledge* of the phase diagram. To implement this criterion, we add a penalizing term to the error functional, which quantifies the lack of confidence at intermediate \bar{h} values. Here, we simply choose the Shannon entropy applied to the network output, since the result of the Softmax activation function can be interpreted as a probability distribution.

The full error functional used here for training the network to determine the spin chain phase diagram then reads

$$\begin{aligned} \text{Cost}(\hat{f}, f) = & - \sum_{x \in \text{TD}} \sum_i^2 f_i(x) \ln \hat{f}_i(x) \\ & - \delta \sum_{x \in \text{TR}} \sum_i^2 \hat{f}_i(x) \ln \hat{f}_i(x) + \mu |V|^2, \end{aligned} \quad (8)$$

where TD stands for training data, i.e., entanglement spectra from $\bar{h} = 0.25$ and 12.0 , while TR stands for transition region, i.e., entanglement spectra at intermediate disorder strengths

$0.25 < \bar{h} < 12.0$. We stress once again that the set for TR is not labeled, meaning we do not make *any* assumption about the nature of the states in the TR. In the last two terms of Eq. (8), δ and μ are further hyperparameters controlling the importance of confidence optimization and the strength of weight decay, respectively. We choose suitable values empirically by requiring optimal minimization of the error on the *testing* data. In particular, we observe that as long as both μ and δ are chosen to be of order 1, their exact values do not influence the results significantly. For the following applications, we therefore choose $\delta = \mu = 1$, unless otherwise noted. To understand the influence of the respective terms, see Fig. 8 in Appendix C, where the phase transition regions obtained from networks trained with all possible combinations of $\delta, \mu \in \{0, 1\}$ are compared.

We refer the reader to Appendix B for further information on the hyperparameters used in Eq. (8) and to Appendix C for comparison of results for different system sizes. In all cases, there is no fine-tuning of the network needed. We have checked that changing the hyperparameters slightly from the values we used does not induce noticeable variations in the classification output.

V. RESULTS AND COMPARISON WITH CONVENTIONAL METHODS

A. Disorder-averaged phase diagram

With the single-hidden-layer neural network described above, we were able to reproduce the phase diagram of the model given by Eq. (5). Figure 1 shows the confidence for the MBL phase averaged over 40 disorder realizations of the $N = 16$ chain as a function of the field \bar{h} and the energy density. A quantitative determination of the critical value of \bar{h} that corresponds to the transition between the ETH and MBL regimes is in part a question of definition. In order to define a critical \bar{h}_c , there are two quantities that need to be specified: the threshold for the network confidence, above which a given entanglement spectrum is classified as being in the MBL regime, and the fraction of eigenstates that need to be classified as MBL by lying above this threshold. We show in Fig. 3(a) the resulting dependence of \bar{h}_c on these two quantities for the $N = 16$ chain. For example, if we consider states above a threshold of 90% confidence as being MBL, and require that half of all spectra belonging to a certain value of \bar{h} have to be classified as MBL by this criterion to be at the transition, we obtain a critical value of about $\bar{h}_c = 2.8 \pm 0.5$. This agrees with the literature [18,36,43].

B. Single disorder realization

To compare the performance with conventional methods in more detail, we consider a specific disorder realization instead of averaging over many. Note, however, that we nevertheless average separately over multiple training runs for cross-validation as explained in Sec. II. This is also required from the observation of slight deviations in the classification of single eigenstates when the network is trained multiple times, even when this is done with the same input data and training parameters. These deviations can be traced back to the randomized weight and bias initialization we use. See

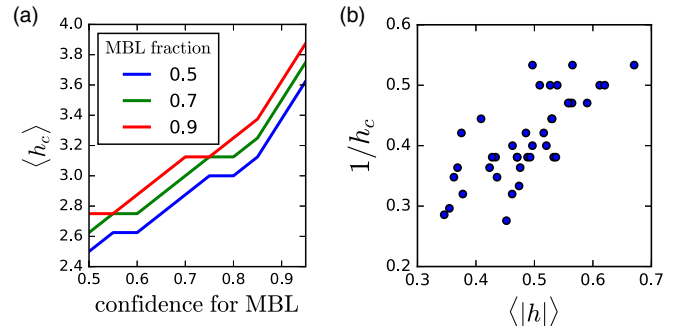


FIG. 3. (a) Dependence of the critical value \bar{h}_c on how confident the network is required to be in classifying a given entanglement spectrum as MBL for the average of 40 disorder realizations of the $N = 16$ chain. Here, different lines denote different percentages of MBL spectra that are required in order to classify a given \bar{h} as MBL. For the transition value \bar{h}_c , we then take the smallest \bar{h} that is classified as MBL in this way. The plateaus come from the finite \bar{h} resolution $\Delta\bar{h} = 0.125$. (b) Correlation of the critical values \bar{h}_c obtained from individual disorder realizations with respective mean disorder strength $\langle |h| \rangle$, averaged over all sites of the $N = 16$ chain, for 40 individual disorder realizations. The correlation coefficient in this case is $\rho = \text{cov}(\bar{h}_c, \langle |h| \rangle) / [\sigma(\bar{h}_c)\sigma(\langle |h| \rangle)] \approx 0.76$, with cov denoting the covariance, and σ the standard deviation, respectively.

Appendix C for a quantitative analysis of this (on average negligibly small) deviation.

To obtain a phase diagram as a function of \bar{h} for a single disorder realization $\{h_r\}$, we generate $\{h_r\}$ for $\bar{h} = 1$, and then rescale it as $\{\bar{h}h_r\}$. Figure 4(a) shows the standard deviation of the entanglement entropy over 512 consecutive eigenstates of the $N = 18$ chain. The entanglement entropy is expected to be larger in the volume-law entangled ETH regime than in the area law entangled MBL regime. Thus, in the transition region, where some entanglement spectra are MBL-like and some are ETH-like, the entanglement entropy will vary most strongly from one eigenstate to the next [39–41,43]. The maximum in the variance can thus be associated with the transition between the two regimes.

Figure 4(c) shows the number of states classified as neither ETH nor MBL for each individual eigenstate of the same system, averaged over 512 consecutive eigenstates. The sharply defined region where this uncertainty is maximal can be interpreted as the neural network's estimate for the transition between ETH and MBL regimes. Note that to obtain this figure, we also performed cross-validation over 50 training runs.

We compare this result to two established criteria to determine the phase transition, which are well defined for a single disorder realization (and hence also for a fixed system size). Figure 4(a) shows the standard deviation of the von Neumann entanglement entropy. Figure 4(b) shows the absolute value of the \bar{h} -derivative of the Schmidt gap for the same system. As the Schmidt gap is small in the ETH phase and large in the MBL phase, its \bar{h} -derivative is expected to attain its maximum value at the transition [50]. Comparing Figs. 4(a)–4(c), we find that the shape and location of the transition agree very well. At the same time, the neural network pins down a much sharper transition than the other approaches.

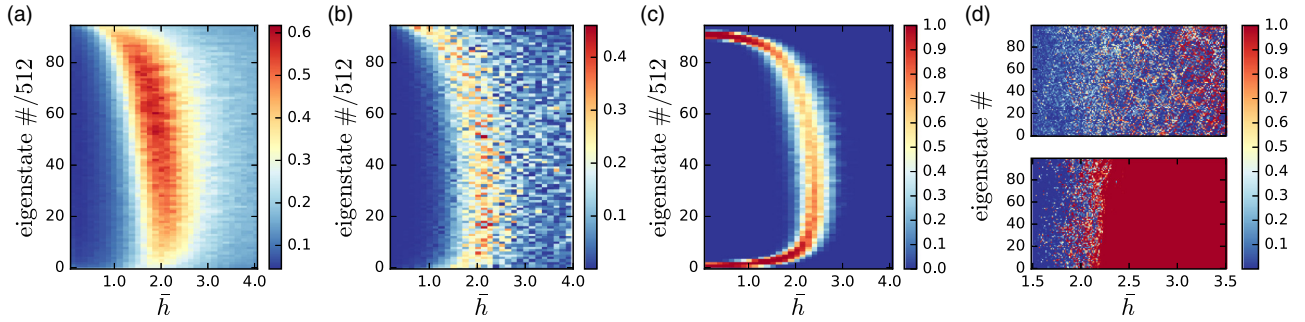


FIG. 4. Comparison of energy-resolved single-sample ETH to MBL transition indicators. (a) Standard deviation of the von Neumann entanglement entropy of a cut of length $N_A = 9$ of the $N = 18$ chain over 512 consecutive eigenstates. (b) Absolute value of the \bar{h} -derivative of the Schmidt gap for the same system averaged over 512 consecutive eigenstates. (c) Uncertainty in the classification of entanglement spectra by a neural network with one hidden layer, including cross-validation over 50 trainings. States classified as MBL with a confidence larger than 0.1 but smaller than 0.9 are assigned a 1, all others a 0. The continuous color range comes from averaging over 512 consecutive eigenstates. (d) Fine structure comparison of the classification of 100 representative eigenstates taken from the middle of the spectrum of a single sample of the $N = 16$ chain (where $N_A = 8$). The upper panel shows the value of Schmidt gap, while the lower panel shows the confidence of the neural network for classifying a given state as MBL. Note that this fine-structure analysis cannot be performed with the entanglement entropy standard deviation criterion, as this would require a coarse-graining of eigenstates.

The power of the neural network based method is even more evident, when the classification of individual eigenstates is considered. Figure 4(d) shows that the classification of individual states as belonging to either the ETH or the MBL regime by the neural network is much clearer than by the Schmidt gap criterion.

Next, we compare the neural network based transition characterization with the established method of energy level statistics (see Sec. III). To identify the energy level statistics, we use the average ratio r of adjacent energy gaps. For an ordered spectrum $\{E_n; E_n \leq E_{n+1}\}$, the ratio of adjacent gaps is defined as

$$r_n = \frac{\min(E_n - E_{n-1}, E_{n+1} - E_n)}{\max(E_n - E_{n-1}, E_{n+1} - E_n)}. \quad (9)$$

Each energy level distribution leads to a given average ratio r of these r_n . The comparison between r and the neural network based transition is shown in Fig. 5. Here it becomes clear that confidence optimization, as represented by an additional term in the cost function [corresponding to $\delta = 1$ in Eq. (8)], which is designed to favor networks which confidently classify transition region states, makes physical sense and is essential to make contact with established methods.

We note that by refining and combining conventional methods [18,36,43,49] the same or better classification success can be achieved. However, here we want to point out that there is no equally simple method, assuming as little prior knowledge, that performs equally well as the machine learning based approach. This approach is very basic: we use a single hidden layer, typical neural activation functions, and apply standard regularization techniques. The only nontrivial input we added to this standard setup is cost optimization, which is effected by a nonzero δ in Eq. (8).

We observe that the location of the transition varies substantially with different disorder realizations sampled from the same distribution characterized by some fixed \bar{h} . The main reason for this is that even with fixed \bar{h} per site of the chain, the mean strength of disorder can still fluctuate. As can be

seen in Fig. 3(b), the disorder strength averaged over all sites $\langle |h| \rangle = \frac{1}{N} \sum_{r=1}^N |h_r|$ is directly correlated with the inverse of the field corresponding to the transition, $1/\bar{h}_c$. We thus identify it as a key ingredient for the dependence of the transition on the disorder realization, even though other properties of the disorder realization than the average absolute field value may also be correlated with the location of the transition.

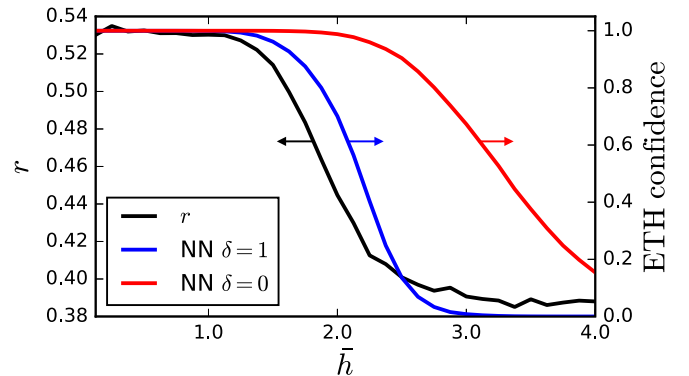


FIG. 5. Comparison of ETH to MBL transition indicators for a single realization of the disorder of the $N = 18$ chain, averaged over energy density. The arrows indicate the vertical axis each curve refers to. The reduced density matrix is built for $N_A = 9$. Here, the ratio r of adjacent energy gaps is colored in black. For the GOE describing the ETH regime, we have $r \simeq 0.530$. For the Poisson level statistics characterizing the MBL regime, we have $r \simeq 0.386$. We compare r to the energy-average of the confidence with which entanglement spectra belonging to a given value of \bar{h} are classified as ETH, obtained from two neural network (NN) instances: one with the confidence-enhancing term we added to the network's cost function, corresponding to $\delta = 1$ in Eq. (8), the other without it, corresponding to $\delta = 0$, colored in blue and red, respectively. The large deviation in the classification of the network trained with $\delta = 0$ from the established criterion of energy level statistics underlines the importance of confidence optimization. Note also that the transition is sharper for the neural network based approach with $\delta = 1$.

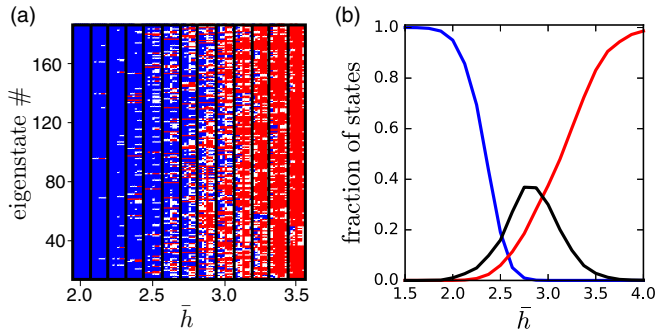


FIG. 6. Dependence of the classification of eigenstates for a single disorder realization of the $N = 18$ chain on the location of the entanglement cut. (a) Close-up of the \bar{h} -dependent classification of 200 eigenstates taken from the middle of the spectrum for entanglement cuts ranging from $N_A = 6$ up to $N_A = 12$. For each value of \bar{h} , a subplot is shown whose horizontal axis corresponds to these seven cuts. Blue (red) states were classified with confidence $> 90\%$ as ETH (MBL). The remaining states are marked in white. (b) Survey of 10000 eigenstates from the middle of the spectrum which have been classified separately for all seven cuts. The blue (red) curve is the fraction of eigenstates that were classified with confidence $> 90\%$ as ETH (MBL) for all cuts. The black curve is the fraction of states that are classified with confidence $> 90\%$ as ETH for at least one cut and with the same confidence as MBL for at least one cut, i.e., states, which are spatially inhomogeneous in their character.

C. Spatial structure of individual eigenstates

The neural network classification of individual eigenstates based on the entanglement spectrum further allows to analyze the *local* structure of these states, by varying the location of the entanglement cut. We compute for a fixed disorder realization the entanglement spectra of each eigenstate for seven consecutive cuts in the middle of the $N = 18$ chain as a function of \bar{h} . All of these entanglement spectra are subsequently classified using the neural network. We do not perform cross-validation as we will only include confidently classified spectra in the subsequent analysis. Due to the variation of the length of the entanglement spectra with the cut location, a new training of the network is necessary for each cut.

Figure 6(a) shows the entanglement cut-resolved classification results for 200 consecutive eigenstates as a function of \bar{h} , by varying the cut for the entanglement spectrum to lie on seven consecutive bonds. The resulting locally resolved classification of eigenstates displays a remarkable asymmetry between approaching the transition from the ETH or from the MBL side; first, we consider the few states that are MBL classified for some entanglement cut deep in the ETH regime. We observe that a substantial fraction of them was classified as MBL consistently across all seven cuts. This is in sharp contrast with the few states that are ETH classified for some entanglement cut deep in the MBL regime. Only a tiny fraction of these states is classified as ETH across all seven cuts simultaneously. Our results point to an asymmetry in the evolution of the local structure of eigenstates when the transition is approached from the MBL or ETH side.

Using this capability of resolving the state character locally, we can thus confirm the following hypothesis about the

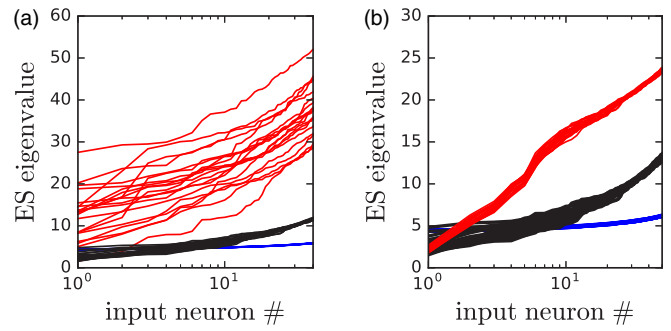


FIG. 7. Dreaming of entanglement spectra for a single disorder realization of the $N = 18$ chain (with $N_A = 9$). An entry on the horizontal axis denotes the index of the respective entanglement spectrum eigenvalue. (a) 140 typical entanglement spectra of the pure ETH (blue), MBL (red), and transition (black) regime. The trained network assigns a confidence vector (1,0) to the ETH spectra, and (0,1) to the MBL spectra. The transition regime entanglement spectra have been chosen such that the network yields an output lying between (0.4,0.6) and (0.6,0.4) on them. (b) Result after 200 000 steps of the dreaming gradient descent, described in the main text, when applied on 20 of the same transition region entanglement spectra. Note the difference in scale, which indicates that the neural network learns relative rather than absolute features of the training data. The network correctly picks up the power-law structure of the entanglement spectrum in the MBL phase, as can be deduced from its nearly linear slope in this logarithmic plot.

evolution of the local structure of the quantum states across the transition [40,46–48]. As the transition is approached from the MBL side, ETH-like regions emerge in rare places where the random field variations are small. These are the first places where quantum fluctuations will dominate over the classical ones induced by the random field. As the transition is approached, these regions grow in size and will eventually be large enough to serve as a bath for the entire system. At this point, the state becomes entirely ETH-like. We make these observations quantitative in Fig. 6(b), where the fraction of pure ETH states, pure MBL states, and of states with mixed character is plotted as a function of \bar{h} .

We remark that for the above analysis to be valid, it is crucial that we are considering spin chains with open boundary conditions. Only in this case does an entanglement cut that separates the chain in two subsystems reveal purely local information about the nature of the many-body eigenstate. In contrast, if the chain was studied with periodic boundary conditions, a bipartitioning of the system requires two cuts and the entanglement spectrum would convolute information about the structure near both cuts at distant locations.

D. What the network has learned: dreaming

Taken collectively, the above results on the disorder-averaged phase diagram as well as about the transition for individual disorder realizations provide strong evidence that the neural network performs the classification operation for which it was designed. To gain further insights into the network operation and to rule out that any pathological behavior emerges, we analyze it with a method called *creation*

by *refinement* [52], which has recently gained widespread attention under the term *dreaming* [53] in the application to image recognition. The method is, however, more general than this, and here we use it on entanglement spectra. For that, the trained network is presented with randomly initialized input data x for which it provides an output y . Subsequently, the input data is modified until the output y equals a desired entry from the target set $\{(1,0),(0,1)\}$, for example $(0,1)$ for being classified as MBL. When this procedure is repeated many times, the obtained collection of input data x reveals which properties of the training data have been learned by the network as being characteristic for an MBL entanglement spectrum.

We perform the dreaming algorithm using gradient descent to optimize the input data for a desired network output, using the cross-entropy between the desired and the actual output as a cost function. Instead of using entirely random input data to initialize the dreaming, we randomly choose entanglement spectra that have not been classified confidently as either MBL or ETH by the network (more precisely, we choose spectra for which the confidence of ETH and MBL is between 40% and 60%). These input spectra are then optimized toward either ETH or MBL. To obtain comparable and physical entanglement spectra, we further enforce the constraints that (i) the eigenvalues of the reduced density matrix be nonnegative and (ii) the absolute value of their logarithm be ordered from smallest to largest. Enforcing these constraints is necessary, because we cannot expect that the network has “learned” any of them, and not restrictive, as any physical entanglement spectrum can be represented this way. Note that we do not require that the eigenvalues of the reduced density matrix sum up to 1, since this property of the training data was lost when restricting to the first half of the entanglement spectrum as described at the beginning of Sec. IV.

To keep the dreaming algorithm from merely capturing structure specific to one training instance of the network, we average over 40 distinct training instances. For each dreaming, we use 2×10^5 steps with a gradient descent stepsize of 10^{-4} . The results are shown in Fig. 7(b) where a comparison with true entanglement spectra from deep in the MBL and deep in the ETH phase is made. We observe from the resemblance between Figs. 7(a) and 7(b) that the neural network has indeed learned the relative characteristic shapes of typical ETH and MBL entanglement spectra, in particular the power-law nature of the entanglement spectra in the MBL phase [37], but without becoming sensitive to their exact absolute magnitudes. This can be understood by noting that it was only trained to distinguish one phase from the other, and not to individually characterize the respective phases on their own.

VI. SUMMARY

We trained an artificial neural network with a single hidden layer on entanglement spectra of the disordered Heisenberg chain to identify the ETH and MBL regimes of this model, and subsequently applied the network to entanglement spectra belonging to states that lie between these two regimes. Even though the network was not trained with entanglement spectra near the phase transition, the phase diagram obtained is in good agreement with previous studies. By adapting the dreaming technique, we were able to show that the network

correctly learns the power-law entanglement spectrum of the MBL phase. Our method is uncontrolled, and is therefore less qualified to deduce a quantitative value for the critical disorder strength, for example. However, it has the advantage of being simpler than conventional methods, and in addition provides the cleanest characterization of the transition in the case of single eigenstates of a single disorder realization.

The structure and operation of the network that we presented, including features such as confidence optimization and dreaming, is broadly applicable to classify other phases of matter based on their entanglement spectrum and likely also other correlation functions as input data. We conclude that artificial neural networks, when used correctly, constitute an efficient and unbiased tool for classifying phases of matter from numerical data.

ACKNOWLEDGMENT

F.S. acknowledges support by the Swiss National Science Foundation under Grant No. 200021-169061.

APPENDIX A: TRAINING WITH OTHER KINDS OF INPUT DATA

In the main text, we have focused on the spectrum of H_e , defined by Eq. (6), as input data for our neural network. In this appendix, we report our observations when training instead with (a) the spectrum of the reduced density matrix ρ_A and (b) the differences of the H_e eigenvalues. The motivation for input data of type (a) is that diagonalizing ρ_A , instead of its logarithm, requires less preprocessing, while the motivation for (b) is that the differences of the H_e eigenvalues have been shown [25,36] to be statistically distributed in a unique fashion depending on whether the regime is MBL or ETH.

In the case of (a), we again are able to train the same network as described in the main text, Eq. (7), to an accuracy of $\eta = 1$. An analysis of the weight distribution of the hidden layer after training shows that for distinguishing between MBL and ETH, the trained network only takes note of the first few input neurons, which correspond to the lowest entanglement eigenvalues. In this case, training has dynamically rediscovered a well known criterion for distinguishing an almost perfectly mixed state (such as a typical ETH state) from a nearly pure state (such as a typical MBL state): the “Schmidt gap” described in Sec. III.

For input data of kind (b), we find that the network described by Eq. (7) does not converge to $\eta = 1$ in any reasonable number of training steps. However, when increasing its complexity by introducing a second hidden layer of neurons, we can again easily achieve $\eta = 1$.

In both cases, (a) and (b), the phase diagram is in quantitative agreement with Fig. 1, as long as both regularization and confidence optimization terms are taken into account in the cost function (8). We note that when we drop either of these additions to the cross entropy, the phase diagram obtained after training the respective network to distinguish MBL and ETH with $\eta = 1$ becomes dependent on the type of input data. This indicates that the naive choice of cost function is not restrictive enough to uniquely determine the neural network parameters. Instead, it has a manifold of minima which yield

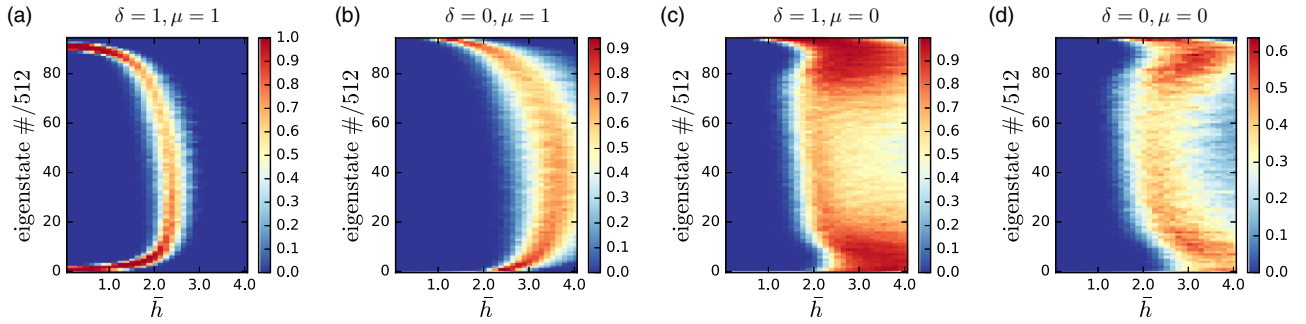


FIG. 8. Comparison of transition regions obtained from neural networks that were trained with different cost functions, corresponding to a different choice of the parameters δ and μ in Eq. (8), for a single sample of the $N = 18$ chain with $N_A = 9$. Here we show the classification uncertainty as defined below Fig. 4(c), i.e., including cross-validation over 50 trainings, for different cost functions. (a) The standard classification result with both regularization and confidence-enhancing terms in the cost function. (b) Without confidence optimization, the transition region is detected at qualitatively larger values of \bar{h} . From this observation alone we cannot decide which of the two networks corresponding to $\delta \in [0, 1]$ should be preferred, since both distinguish the pure ETH and MBL regions with $\eta = 1$. However, Fig. 5 clearly shows that only the cost function with $\delta = 1$ faithfully recovers the transition obtained from a level statistics analysis. In addition, as discussed in Appendix A, when training with different kinds of input data the transition region stays unchanged only when $\delta = 1$, while for $\delta = 0$ the network classification becomes inconsistent. (c) and (d) Without proper regularization, the network performance becomes pathological.

inequivalent phase diagrams. This observation reaffirms our choice of cost function, and in particular motivates the extra criterion of confidence optimization we added to encourage a high-confidence classification of transition states, i.e., the second term in Eq. (8).

APPENDIX B: HYPERPARAMETERS AND NETWORK DETAILS

All neural networks used in this work were implemented in Python using Google's TENSORFLOW [54]. In the main text, it was noted that a network with a large number of hidden layer units needs to be regularized appropriately in order to avoid overfitting. The exact number of hidden layer units is, however, somewhat arbitrary as long as it is large enough with respect to the number of input neurons. For all system sizes considered, with entanglement spectra inputs at most of size $n_1 = \binom{N_A}{\lfloor N_A/2 \rfloor} |_{N_A=9} = 126$, corresponding to a half-cut of the $N = 18$ chain, we chose $n_2 = 1024$ units for the single hidden layer of our network.

During training, we used an empirically determined step-size of 10^{-4} for gradient descent on weights initialized around 0 with a standard deviation of 0.1. Furthermore, we chose the regularization and confidence optimization parameters $\mu = 1$ and $\delta = 1$ in Eq. (7), respectively. We show in Fig. 8 that if either confidence optimization or weight decay is ignored, the classification outcome is not consistent with the results from other methods (see also Fig. 5 in the main text).

For all calculations, we removed the lowest and highest 10% of all eigenvalues for the dataset corresponding to the pure ETH and MBL regimes, which is then subdivided into training and testing set for cross-validation, and train with randomly chosen batches of 100 elements in each step. We use the same batch size for the entanglement spectra coming from transition region states, which are required by the confidence optimization term we added to the cost function. We use between 3000 and 4000 training steps.

APPENDIX C: EMPIRICAL OBSERVATIONS ON THE NETWORK PERFORMANCE

In Sec. V, we noted that the neural network's classification of individual eigenstates for a single disorder realization is not necessarily consistent over multiple training and evaluation runs, even when all hyperparameters and input data are left unchanged. This stems mainly from the random initialization of the weight and bias vectors, not from the randomly chosen training and testing sets. Here we quantify this inconsistency

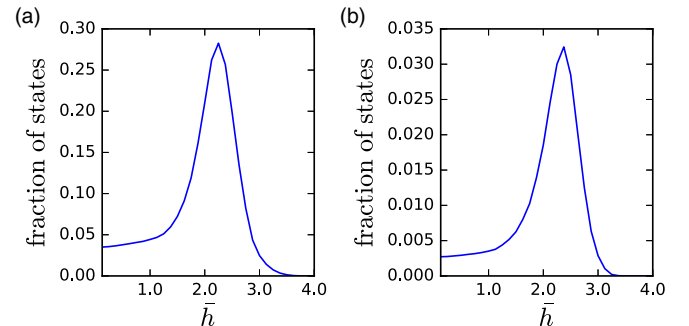


FIG. 9. Dependence of the classification of states on different training runs. For each plot, the entanglement spectra for a system with $N = 18$ sites with a single disorder realization were trained with 50 times, each time with randomly initialized network parameters and a random choice of training data. The resulting classifications of the individual eigenstates for each \bar{h} are then compared pairwise and the comparison is averaged over all pairings of trainings. We consider three categories of states: (i) classified with confidence $> 90\%$ as MBL, (ii) classified with confidence $> 90\%$ as ETH, (iii) others, not confidently classified. (a) Fraction of states that switch between any of the three categories (i)–(iii) to any other between two trainings. (b) Fraction of states that switch from category (i) to category (ii) (i.e., from confidently MBL to confidently ETH) or vice versa between two trainings. Note that the value of \bar{h} at which these fractions assume their maximal value provides one method to determine the critical transition location \bar{h}_c for this particular disorder realization.

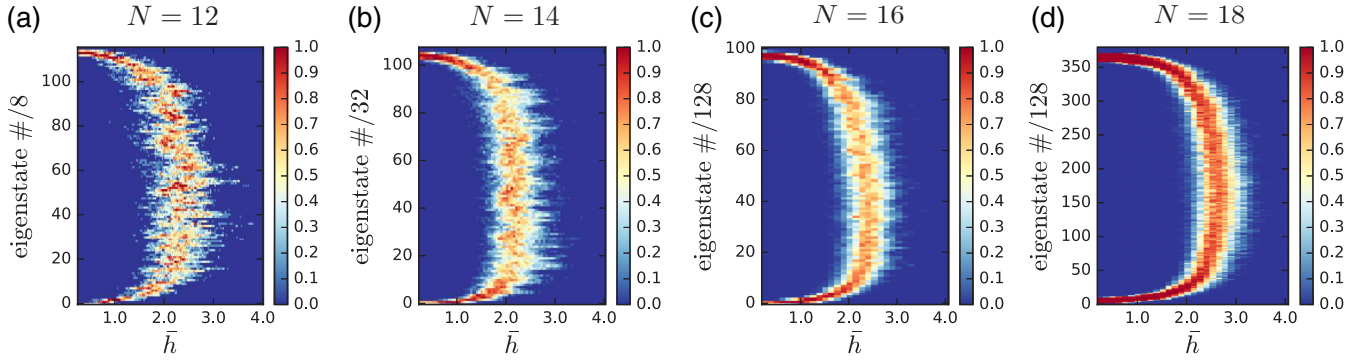


FIG. 10. Finite-size dependence of the transition region for different single disorder realizations of the $N = 12, 14, 16, 18$ chains (note that we have on purpose used a different disorder realization for $N = 18$ than the one used in Fig. 4 to show that the overall shape of the transition region is sample independent). Each plot shows the transition region as defined below Fig. 4(c) for a network trained on entanglement spectra of a cut with $N_A = N/2$. Note that the \bar{h} step size for each plot differs, and is given by $\Delta\bar{h} = 0.01, 0.05, 0.125, 0.125$, respectively. For each N , we have averaged the result over an increasing number of eigenstates in order to arrive at a resolution suited for the human eye in a controllable way. Only for $N = 16$ and 18 , the data are averaged over the same number of eigenstates, in order to demonstrate that the apparent improvement of the sharpness of the transition is not merely a result of a larger number of states that have been averaged over. Note also that each of (a)–(d) necessarily represent different disorder realizations and hence the value of \bar{h} at which the transition occurs should not be compared between them.

in classification for the case of a single-hidden layer network, given by Eq. (7). We trained with entanglement spectra from a single disorder realization of the $N = 18$ chain in 3000 training steps for 50 times. To distinguish between deviations in confidence and deviations in classification, we define an MBL state as a state that is classified as MBL with a confidence of over 90%, and likewise an ETH state as a state which is classified as MBL with a confidence of less than 10% (remember that the confidences for MBL and ETH have to add up to 100% since we use a Softmax output layer). We then train 50 times, and consider all possible pairings of the resulting 50 phase diagrams. It turns out that the fraction of states that change their classification from ETH in one member

of a pair to MBL in the other, or vice versa, averages to 0.008, while the fraction of states that were classified as either MBL or ETH in one member of a pair, but are not confidently classified as either in the other, averages to 0.07. In Fig. 9, we present an \bar{h} -resolved analysis.

We also compare the network performance when trained on entanglement spectra generated from spin chains of different length, as presented in Fig. 10. Here we observe that the transition seems to become sharper for larger system sizes, as expected [40]. However, one has to keep in mind that each panel of Fig. 10 uses a different disorder realization and hence realization-specific features and those stemming from the change in system size both appear in these plots.

-
- [1] L. Zdeborova, *Nat. Phys.* **13**, 420 (2017).
 - [2] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, [arXiv:1608.07848](https://arxiv.org/abs/1608.07848).
 - [3] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, [arXiv:1609.02552](https://arxiv.org/abs/1609.02552).
 - [4] Y. Zhang and E.-A. Kim, *Phys. Rev. Lett.* **118**, 216401 (2017).
 - [5] D.-L. Deng, X. Li, and S. Das Sarma, [arXiv:1609.09060](https://arxiv.org/abs/1609.09060).
 - [6] T. Ohtsuki and T. Ohtsuki, *J. Phys. Soc. Jpn.* **85**, 123706 (2016).
 - [7] J. Carrasquilla and R. G. Melko, *Nat. Phys.* **13**, 431 (2017).
 - [8] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, *Nat. Phys.* **13**, 435 (2017).
 - [9] W. Hu, R. R. P. Singh, and R. T. Scalettar, [arXiv:1704.00080](https://arxiv.org/abs/1704.00080).
 - [10] A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Anropov, C.-Z. Wang, M. J. Kramer, C. Long, and I. Takeuchi, *Sci. Rep.* **4**, 6367 (2014).
 - [11] L. M. Ghiringhelli, J. Vybiral, S. V. Levchenko, C. Draxl, and M. Scheffler, *Phys. Rev. Lett.* **114**, 105503 (2015).
 - [12] S. V. Kalinin, B. G. Sumpter, and R. K. Archibald, *Nat. Mater.* **14**, 973 (2015).
 - [13] L. Li, T. E. Baker, S. R. White, and K. Burke, *Phys. Rev. B* **94**, 245129 (2016).
 - [14] G. Torlai and R. G. Melko, *Phys. Rev. B* **94**, 165134 (2016).
 - [15] G. Carleo and M. Troyer, *Science* **355**, 602 (2017).
 - [16] D.-L. Deng, X. Li, and S. Das Sarma, *Phys. Rev. X* **7**, 021021 (2017).
 - [17] H. Li and F. D. M. Haldane, *Phys. Rev. Lett.* **101**, 010504 (2008).
 - [18] D. J. Luitz, N. Laflorencie, and F. Alet, *Phys. Rev. B* **91**, 081103 (2015).
 - [19] P. W. Anderson, *Phys. Rev.* **109**, 1492 (1958).
 - [20] L. Fleishman and P. W. Anderson, *Phys. Rev. B* **21**, 2366 (1980).
 - [21] I. V. Gornyi, A. D. Mirlin, and D. G. Polyakov, *Phys. Rev. Lett.* **95**, 206603 (2005).
 - [22] D. Basko, I. Aleiner, and B. Altshuler, *Ann. Phys.* **321**, 1126 (2006).
 - [23] D. A. Huse, R. Nandkishore, and V. Oganesyan, *Phys. Rev. B* **90**, 174202 (2014).
 - [24] R. Nandkishore and D. Huse, *Annu. Rev. Condens. Matter Phys.* **6**, 15 (2015).

- [25] S. D. Geraedts, R. Nandkishore, and N. Regnault, *Phys. Rev. B* **93**, 174202 (2016).
- [26] M. Serbyn, Z. Papić, and D. A. Abanin, *Phys. Rev. Lett.* **111**, 127201 (2013).
- [27] B. Swingle, [arXiv:1307.0507](https://arxiv.org/abs/1307.0507).
- [28] V. Ros, M. Müller, and A. Scardicchio, *Nucl. Phys. B* **891**, 420 (2015).
- [29] D. A. Huse, R. Nandkishore, V. Oganesyan, A. Pal, and S. L. Sondhi, *Phys. Rev. B* **88**, 014206 (2013).
- [30] Y. Avishai, J. Richert, and R. Berkovits, *Phys. Rev. B* **66**, 052416 (2002).
- [31] V. Oganesyan and D. A. Huse, *Phys. Rev. B* **75**, 155111 (2007).
- [32] A. Pal and D. A. Huse, *Phys. Rev. B* **82**, 174411 (2010).
- [33] R. Modak and S. Mukerjee, *New J. Phys.* **16**, 093016 (2014).
- [34] K. Agarwal, S. Gopalakrishnan, M. Knap, M. Müller, and E. Demler, *Phys. Rev. Lett.* **114**, 160401 (2015).
- [35] Y. Bar Lev, G. Cohen, and D. R. Reichman, *Phys. Rev. Lett.* **114**, 100601 (2015).
- [36] Z.-C. Yang, C. Chamon, A. Hamma, and E. R. Mucciolo, *Phys. Rev. Lett.* **115**, 267206 (2015).
- [37] M. Serbyn, A. A. Michailidis, D. A. Abanin, and Z. Papić, *Phys. Rev. Lett.* **117**, 160601 (2016).
- [38] B. Bauer and C. Nayak, *J. Stat. Mech.* (2013) P09005.
- [39] J. A. Kjäll, J. H. Bardarson, and F. Pollmann, *Phys. Rev. Lett.* **113**, 107204 (2014).
- [40] V. Khemani, S. P. Lim, D. N. Sheng, and D. A. Huse, *Phys. Rev. X* **7**, 021013 (2017).
- [41] R. Singh, J. H. Bardarson, and F. Pollmann, *New J. Phys.* **18**, 023046 (2016).
- [42] C. Monthus, *Entropy* **18**, 122, (2016).
- [43] X. Yu, D. J. Luitz, and B. K. Clark, *Phys. Rev. B* **94**, 184202 (2016).
- [44] W. De Roeck, F. Huveneers, M. Müller, and M. Schiulaz, *Phys. Rev. B* **93**, 014203 (2016).
- [45] P. Prelovšek, O. S. Barišić, and M. Žnidarič, *Phys. Rev. B* **94**, 241104 (2016).
- [46] R. Vosk, D. A. Huse, and E. Altman, *Phys. Rev. X* **5**, 031032 (2015).
- [47] A. C. Potter, R. Vasseur, and S. A. Parameswaran, *Phys. Rev. X* **5**, 031033 (2015).
- [48] L. Zhang, B. Zhao, T. Devakul, and D. A. Huse, *Phys. Rev. B* **93**, 224201 (2016).
- [49] M. Serbyn, Z. Papić, and D. A. Abanin, *Phys. Rev. X* **5**, 041047 (2015).
- [50] J. Gray, S. Bose, and A. Bayat, [arXiv:1704.00738](https://arxiv.org/abs/1704.00738).
- [51] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)* (Springer-Verlag, New York, Secaucus, NJ, USA, 2006).
- [52] J. P. Lewis, in *IEEE 1988 International Conference on Neural Networks* (IEEE, San Diego, CA, USA, 1988), pp. 229–233 Vol. 2.
- [53] A. Mordvintsev, C. Olah, and M. Tyka, Inceptionism: Going deeper into neural networks, Google Research Blog. Retrieved June 20 (2015), p. 14.
- [54] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TENSORFLOW: Large-Scale Machine Learning on Heterogeneous Systems* (2015), software available from tensorflow.org.